

INDEX

A

- Active Directory
 - attacks
 - DCSync, 82–83
 - Golden Ticket, 82–83
 - course, Hack the Box, 235
 - active information gathering, 18–19
 - port scanning
 - with Metasploit, 25–26
 - with Nmap, 19–25
 - add_group_user command, 82, 240
 - Address Resolution Protocol (ARP)
 - scans, 209–211
 - add_user command, 82, 240
 - administrator-level procedures,
 - enabling, 174–175
 - administrators, impersonating, 243
 - Advanced Network Scans, in Nessus,
 - 43–44
 - A flag, Nmap port scanning, 20
 - Aircrack-ng suite, 132–133
 - aircrack-ng tool, 136
 - aireplay-ng tool, 134
 - Airgeddon, 136–138
 - airmon-ng tool, 132–133
 - airodump-ng tool, 132–133, 135
 - Alfa Network, 132
 - Amazon Simple Storage Service (S3)
 - buckets, 29, 222–223
 - Amazon Web Services (AWS), 220–222,
 - 225–226
 - AMD64 architecture, 234–235
 - anonymous logins, 30
 - anonymous module, 30
 - antivirus evasion, 91–92
 - creating binaries with MSFvenom,
 - 92–93
 - custom executable templates,
 - 97–98

- developing custom payloads,
 - 101–104
 - encoding with MSFvenom, 93–96
 - evasion modules, 99–101
 - generating executables from
 - Python files, 104–105
 - launching payloads stealthily, 98–99
 - packing executables, 96–97
- Apache Tomcat, attacking, 212–214
- Apple Silicon architecture, 235–236
- arch label, module info sheet, 54
- ARM architecture, 235–236
- Armitage, 10
- ARP (Address Resolution Protocol)
 - scans, 209–211
- assembly language basics, 156
- automated vulnerability scanning. *See*
 - vulnerability scanning
- AutoPwn2* module, 125–126
- autoroute* module, 208
- auxiliary modules, 8, 145
 - categories of, 146
 - creating, 149–154
 - debugging, 153–154
 - running modules, 151–153
 - writing modules, 149–151
 - listing all available, 146–147
 - searching for HTTP modules,
 - 148–149
 - using, 147–148
- Auxiliary::Scanner mixin, 32
- available targets, module info sheet, 55
- Awesome AWS S3 Security git
 - repository, 223
- Az-Blob-Attacker tool, 231

B

- background command, 78–79, 240, 243
- backward jumps, 197–198

- bad characters, identifying, 198–201
- banner grabbing, 36–37, 211–212
- Bash Bunny, 118–119
- Basic Service Set Identifier (BSSID), 133
- binaries, creating with MSFvenom, 92–93
- bind shells, 8
- bleeding edge repositories, 108
- browser-based exploits, 122–123
 - automating with *AutoPwn2*, 125–126
 - finding in Metasploit, 123–125
 - finding more recent, 126
- brute-force attacks, 68–69, 212–214
- buffer overflows, porting, 157–159
 - adding randomization, 163
 - configuring exploit definitions, 160–161
 - implementing features of Framework, 162
 - removing dummy shellcode, 164–166
 - removing NOP slides, 163–164
 - stripping existing exploits, 159–160
 - testing base exploits, 161–162
- bypassing two-factor authentication, 116–117

C

- carriage returns and line feeds (CRLFs), 182
- cheat sheet, 239–243
- check command, 239
- Chen, Wei, 49
- childProcess function, 224
- clearev command, 89, 240
- clearing logfiles, 89
- client-side attacks, 121–122
 - browser-based exploits, 122–126
 - file-format exploits, 126–128
- clone system call, 224
- cloning websites, 113–117
- cloud
 - functions, 221
 - security, 219–220
 - container takeovers, 226–229
 - Docker containers, 223–225
 - escaping Docker containers, 229–231
 - identity and access management
 - tools, 220
 - Kubernetes, 231
 - serverless functions, 221–222
 - setting up cloud testing environments, 225–226
 - storage, 222–223

- CloudGoat
 - container takeovers, 226–229
 - escaping Docker containers, 229–231
 - setting up, 225–226
- cloud look and bypass* module, 18
- Cobalt Strike, 10
- Cognito, 220
- commands
 - injection, 214–215
 - Meterpreter, 240–242
 - post exploitation, 242–243
 - MSFconsole, 239–240
 - MSFvenom, 242
- Common Vulnerabilities and Exposures (CVE)
 - IDs, 52–53
- community strings, 31
- compressed executables, 96–97
- compromising Windows virtual machines, 67–70
- containers, Docker, 223–225
 - CloudGoat, 225–226
 - container takeovers, 226–229
 - escaping, 229–231
 - Kubernetes, 231
- covert penetration testing, 5
- credentials, harvesting, 139–143
- creds_all command, 75–76
- CRLFs (carriage returns and line feeds), 182
- custom
 - executable templates, 97–98
 - payloads, developing, 101–104
 - scanners, writing, 32–33

D

- data centers, 219
- data execution prevention (DEP), 157, 201
- db_connect command, 239
- db_create command, 239
- db_destroy command, 239
- db_import command, 21, 40, 45
- db_nmap command, 24, 239
- db_status command, 21
- DCSync attacks, 82–83
- dcsync_ntlm command, 76
- deauthentication (deauth) attacks, 133–135
- deauthentication frames, 133–134
- debugging
 - auxiliary modules, 153–154
 - buffer overflows, 161–162

- SEH overwrite, 196
 - testing fuzzers, 188–191
- deepce* script, 231
- DEP (data execution prevention), 157, 201
- description section, module info sheet, 55
- detection evasion, 91–92
 - creating binaries with MSFvenom, 92–93
 - custom executable templates, 97–98
 - developing custom payloads, 101–104
 - encoding with MSFvenom, 93–96
 - evasion modules, 99–101
 - generating executables from Python files, 104–105
 - launching payloads stealthily, 98–99
 - packing executables, 96–97
- developer certificates, custom payloads with, 101–104
- dictionary attacks, 136
- digest authentication, 76
- dig* tool, 18
- dir* command, 62
- disabling protections, 156–157
- Discord community, 233
- DistCC, 214–215
- Docker containers, 223–225
 - CloudGoat, 225–226
 - container takeovers, 226–229
 - escaping, 229–231
 - Kubernetes, 231
- Docker Desktop application, installing, 235–236
- docker-escape* tool, 231
- docker info* command, 229–230
- documenting intelligence gathering, 16
- domain administrator tokens, stealing, 242
- domain controllers
 - DCSync attacks, 82–83
 - Golden Ticket attacks, 82–83
 - token impersonation, 80–82
- Domain Name System (DNS), 16–18
- DoS attacks, 133–135
- dos/wifi/deauth* modules, 149
- download* command, 241
- downloading Nessus scan reports, 47
- drop_token* command, 241
- Ducky Script, 118
- dummy shellcode, removing, 164–166, 169–170
- dumping password hashes, 242–243

E

- egg hunting, 192
- EIP (extended instruction pointer) registers, 156
- Elasticsearch application, 206–207
- email
 - malicious, 110–112
 - server setup, 109–110
- encoders, 12
- encoding with MSFvenom, 93–96
 - enum_s3* module, 222–223
- escaping Docker containers, 229–231
- ESP (extended stack pointer) registers, 156
- ESSID (Extended Service Set Identifier), 133
- ethics, xxvii
- evasion modules, 99–101
- evasion techniques, 91–92
 - creating binaries with MSFvenom, 92–93
 - custom executable templates, 97–98
 - developing custom payloads, 101–104
 - encoding with MSFvenom, 93–96
 - generating executables from Python files, 104–105
 - launching payloads stealthily, 98–99
 - packing executables, 96–97
- event logs, clearing, 216–217
- event_manager* tool, 216–217
- Evilginx, 108, 116–117
- Evilgophish, 108, 117
- Evil Portal* module, 141–142
- Evil Twin attacks, 136–138
- executables
 - creating binaries with MSFvenom, 92–93
 - custom templates, 97–98
 - developing custom payloads, 101–104
 - embedding payloads in, 98–99
 - encoding with MSFvenom, 94–96
 - generating, 11
 - from Python files, 104–105
 - packing, 96–97
- execute commands, 241
- exploitation, 3, 51. *See also* porting exploits to Metasploit
 - basic, 52
 - client-side attacks, 121–122
 - browser-based exploits, 122–126
 - file-format exploits, 126–128
 - searching for exploits, 52–55

exploitation (*continued*)
 SEH overwrites
 adding payloads, 198
 bad characters in, 198–201
 creating exploits, 192–194
 getting return addresses, 194–196
 including backward jumps and near jumps, 197–198
 selecting exploits, 56–59
 simulated penetration tests, 206–208
 of Ubuntu machine, 63–65
 of Windows machine, 60–63
exploit command, 59, 64, 79, 180, 240
exploit modules, 8
exploits, defined, 8
exporting Nessus scans, 47
Extended Service Set Identifier (ESSID), 133
extracting password hashes, 72–73

F

false negatives, 37
false positives, 37, 44
file-format exploits, 126–128
fingerprinting, 5
forensic analysis, thwarting, 215–216
framework.log file, 153
FTP servers, scanning for, 30
ftp_version module, 30
fuzzing, 185–186
 developing SEH overwrites
 adding payloads, 198
 creating exploits, 192–194
 getting return addresses, 194–196
 including backward jumps and near jumps, 197–198
 identifying bad characters, 198–201
 identifying vulnerabilities
 controlling SEH, 190–192
 downloading test applications, 186
 testing fuzzers, 187–190
 writing fuzzers, 186–187
fuzz strings, 187

G

gadgets, 157–158
Game of Active Directory lab environment, 234
Gates, Chris, 149
gather modules, 85

`generate_seh_payload` function, 169
generating executables from Python files, 104–105
`getprivs` command, 241
`getsystem` command, 78, 241
`getuid` command, 78–79
 _GNU_SOURCE, 223–224
Golden Ticket attacks, 82–83
`golden_ticket_create` command, 83
Google Safe Browsing API, 149–153
Gophish, 108, 112–113

H

h2b (hex-to-binary) conversion, 181–182
Hack the Box online Active Directory course, 235
Hak5, 139
handshakes, capturing and cracking, 135–136
harvesting
 credentials, 139–143
 usernames and passwords, 113–116
`hashdump` command, 241
hashes
 dumping, 242–243
 extracting, 72–73
 Golden Ticket attacks, 82–83
 Mimikatz, 75–76
 pass-the-hash technique, 74
`help` command, 41, 241
hex blobs, 174
hex-to-binary (h2b) conversion, 181–182
`hosts` command, 21–22, 39, 45, 48
HttpClient mixin, 150
http://flaws.cloud site, 29
HTTP modules, searching for, 148–149
HTTP PUT method, 212
human interface devices (HIDs), 118–119

I

identity, 220
identity and access management (IAM) tools, 220
images, Docker, 223
IMAP (Internet Message Access Protocol)
 fuzzer, 186–187
Immunity Debugger
 buffer overflows, 161–162
 SEH overwrites, 196
 testing fuzzers, 188–191
`impersonate_token` command, 241

- impersonating administrators, 243
- importing
 - Nessus results into Metasploit, 45–46, 48
 - Nexpose reports into Metasploit, 40
 - Nmap scan results into Metasploit, 20–22
- incognito command, 81–82, 243
- indirect information gathering techniques, 16–18
- Infectious Media Generator attacks, 117–119
- info command, 54, 148, 194, 240
- intelligence gathering, 2, 15
 - active information gathering, 18–19
 - port scanning with Metasploit, 25–26
 - port scanning with Nmap, 19–25
 - custom scanners, writing, 32–33
 - documenting, 16
 - passive information gathering, 16–18
 - simulated penetration tests, 204–205
 - targeted scanning, 26
 - for FTP servers, 30
 - for poorly configured MS SQL servers, 28–29
 - for S3 buckets, 29
 - for Server Message Block, 26–28
 - for Simple Network Management Protocol, 31–32
 - for SSH server versions, 29–30
 - test environments, operating in, 16
- interactive remote GUIs, accessing, 84
- interfaces, Metasploit, 9–10
- Internet Message Access Protocol (IMAP)
 - fuzzer, 186–187
- IP address of servers
 - finding, 17–18
 - TCP idle scans, 22–24
- ipconfig command, 208
- irb shell, 89

J

- Jenkins server, 60–61
- jmp esp command, 113–116
- JMP ESP instructions, 156–158, 162

K

- Kali Linux
 - downloading, 234
 - installing, 5–6
 - metapackages, 236–237

- password dictionary, 69
- setting up lab environments, 233–237
- Whois lookup, 16–17
- Karma attacks, 136
- Kelley, Josh, 173
- Kennedy, David, 107, 173
- kerberos_ticket_use command, 83
- Kerberos tokens, 80–82
- keyscan_dump command, 241
- keyscan_start command, 241
- keyscan_stop command, 241
- keystroke logging, 71–72, 243
- kiwi module, 75–76, 83
- krbtgt (Kerberos Ticket Granting Ticket), 82
- Kubernetes Goat, 231

L

- lab environments, setting up, 233
 - ARM and Apple Silicon, 235–236
 - installing Kali metapackages, 236–237
 - x86 and AMD64, 234–235
- lambda functions, 221–222
- lateral movement techniques, 80–83
- LHOST option, 240
- Linux system
 - establishing persistence on, 85–88
 - scanning, 209–211
 - setting up lab environments, 233–237
- listeners, 8, 10
- list_tokens -g command, 241
- list_tokens -u command, 241
- list_tokens -u function, 81
- loadpath command, 187
- local_exploit_suggester module, 79
- log4j vulnerability, 53
- Log4Shell HTTP Header injection exploits
 - info sheet for, 54–56
 - running, 59
 - saving settings, 59
 - selecting, 56–57
 - setting/unsetting options and parameters, 58–59
 - showing payloads for, 57–58
 - showing targets, 58
- logfiles, clearing, 89
- login pages, harvesting usernames and passwords from, 113–116

logins, anonymous, 30
ls -al command, 229
ls command, 241

M

MailCarrier exploits, 157–159
 adding randomization, 163
 configuring exploit definitions, 160–161
 implementing features of Framework, 162
 removing dummy shellcode, 164–166
 removing NOP slides, 163–164
 stripping existing exploits, 159–160
 testing base exploits, 161–162
mail exchange (MX) records, looking for, 18
make_nops function, 163
malicious email, sending, 110–112
man-in-the-middle attacks, 116–117, 138–139
mdk4 tool, 134–135
Memelli, Matteo, 186
memory-resident attacks, 215–216
metapackages, Kali, 236–237
Metasploit, xxiv, 7
 installing, 5–6
 interfaces, 9–10
 Pro, 7, 13
 terminology, 8
 utilities, 11–12
Metasploitable, installing, 5–6
Meterpreter, 67
 basic commands, 70–71
 capturing keystrokes, 71–72
 capturing screenshots, 71
 commands for, 84–88, 240–242
 compromising Windows machines, 62–63, 67–70
 DCSync attacks, 82–83
 developing custom payloads, 101–104
 enabling Remote Desktop Services, 84
 establishing persistence, 85–88
 extracting password hashes, 72–73
 finding platform information, 71
 Golden Ticket attacks, 82–83
 lateral movement techniques, 80–83
 manipulating Windows APIs with Railgun, 88–89
 Mimikatz and *kiwi*, 75–76
 pass-the-hash technique, 74
 pivoting, 89

 post-exploitation commands and syntax, 242–243
 privilege escalation, 77–80
 scraping systems, 85
 token impersonation, 80–82
 viewing all traffic on targets, 84–85
microservices architecture, 231
Microsoft certificates, custom payloads with, 101–104
Microsoft SQL Server
 creating MS SQL modules, 178
 defining exploits, 180
 editing existing modules, 178–179
 running exploits, 183–184
 running shell exploits, 180
 uploading PowerShell scripts, 181–183
Express, installing, 234–235
 getting command execution on existing modules, 173–178
 scanning for poorly configured, 28–29
migrate command, 72, 241
migration, 71, 243
Mimikatz, 75–76
mixins, 32, 150
mobile device attacks, 119
modules, 8, 173. *See also* auxiliary modules
 creating, 178
 defining exploits, 180
 editing existing modules, 178–179
 running exploits, 183–184
 running shell exploits, 180
 uploading PowerShell scripts, 181–183
 getting command execution on MS SQL, 173–178
 info sheets, 54–55
 running, 59
 saving settings, 59
 searches for, 53–54
 narrowing, 52
 selecting, 56–57
 setting/unsetting
 options, 58–59
 parameters, 59
 showing payloads for, 57–58
 showing targets, 58
 /modules/auxiliary directory, 146
 module side effects section, module info sheet, 55
 mona.py file, 162

- !mona seh command, 169
- monster-in-the-middle attacks, 116–117, 138–139
- Moore, H.D., xxiv
- Morales, Antonio, 201
- MSFconsole, 9
 - frequently used commands and syntax, 239–240
 - help files, accessing, 9
 - launching, 9
 - running Nexpose in, 40–42
 - running Nmap from, 24–25
- msf/core* gem, importing, 149
- MSFvenom, 11–12
 - creating and encoding payload, 242
 - creating stand-alone binaries with, 92–93
 - encoding with, 93–96
- mssql_exec* auxiliary module, 175–176
- mssql_payload* exploit, 178
- mssql_ping* module, 28–29
- mssql_powershell* module, 173–174
- mssql_powershell.rb* exploit, 178, 180
 - editing existing module, 178–179
 - running, 183–184
 - shell, 180
 - uploading PowerShell scripts, 181–183
- mssql.rb* file, 176–177, 181, 183
- mssql_xpcmdshell_enable* function, 177
- mssql_xpcmdshell* function, 176–177
- Mudge, Raphael, 10
- multi/handler* module, 92–93, 103
- multi/http/tomcat_mgr_deploy* exploit, 212–214
- multi/manage/autoroute* module, 209
- MX (mail exchange) records, looking for, 18
- mysql_login* module, 68–69
- MySQL servers, brute-forcing authentication on, 68–69

N

- nasm_shell.rb* utility, 12
- NAT (Network Address Translation), 25–26
- near jumps, 197–198
- Nessus, 42
 - Bridge plug-ins, 46
 - configuring, 42–43
 - creating scans, 43–44
 - false positives, reducing, 44
 - importing results into Metasploit, 45–46

- scanning in Metasploit, 46–48
 - viewing reports, 44–45
- nessus_db_import* command, 48
- nessus_scan_download* command, 47
- nessus_scan_export* command, 47
- nessus_scan_launch* command, 47
- nessus_scan_list* command, 47
- Netcat, 36–37
- Netcraft, 17
- netstat commands, 235
- Network Address Translation (NAT), 25–26
- Nexpose, 37
 - configuring, 37–40
 - home screen, 38
 - importing reports into Metasploit, 40
 - New Report Wizard, 39–40
 - New Site Wizard, 39
 - running in MSFconsole, 40–42
- next SEH (NSEH) records, 192, 197
- ngrok TCP proxy, 228–229
- Nmap, port scanning with
 - compromising Windows virtual machines, 68
 - container takeovers, 227
 - importing results into Metasploit, 20–22
 - running Nmap from MSFconsole, 24–25
 - running quick scans, 19–20
 - service-enumeration scans, 20
 - simulated penetration tests, 204–205
 - TCP idle scans, 22–24
- nmap command, 59
- NOP (no-operation) instructions, 156, 169
 - slides, 156, 163–164

O

- obscure services, attacking, 214–215
- O.MG Cable, 119
- opcodes, 12
- open source intelligence (OSINT), 16
- OptBool.new* function, 150
- options, setting/unsetting, 58–59
- overt penetration testing, 4–5

P

- packet recorder, 84
- packing executables, 96–97
- parameters, setting/unsetting, 59
- passive information gathering, 16–18
- pass-the-hash technique, 74

- passwords
 - dictionary, 69
 - dumping hashes, 242–243
 - extracting hashes, 72–73
 - Golden Ticket attacks, 82–83
 - harvesting with Zphisher, 113–116
 - keystroke logging, 71–72
 - pass-the-hash technique, 74
- payload.encoded function, 164
- payloads, 8
 - developing custom, 101–104
 - encoding with MSFvenom, 94–96
 - launching stealthily, 98–99
 - sending malicious email, 110–112
 - viewing active lists of, 57
- PEASS (Privilege Escalation Awesome Scripts Suite), 230
- Penetration Testing Execution Standard (PTES), xxiv, 1
 - covert tests, 5
 - installing Kali, Metasploit, and Metasploitable, 5–6
 - overt tests, 4–5
 - phases of, 2–4
 - vulnerability scanners, 5
- PercussiveElbow, 231
- permissions, 220, 242
- persistence, establishing, 85–88, 207–208
- pfSense firewall, 234
- phases of penetration testing, 2–4
- phishing attacks, 109–113
- ping command, 19
- Piper, Scott, 29
- pivoting, 25, 89
- platform information, finding, 71
- platform labels, module info sheet, 54
- PMFs (protected management frames), 134
- Pn flag, Nmap port scanning, 19
- policies, 220
- polymorphic encoders, 96
- PolyPack project, 97
- POP-POP-RETN instruction pointer, 166, 169, 192–198
- porting exploits to Metasploit, 155–156
 - assembly language basics, 156
 - buffer overflows, 157–159
 - adding randomization, 163
 - configuring exploit definitions, 160–161
 - implementing features of Framework, 162
 - removing dummy shellcode, 164–166
 - removing NOP slides, 163–164
 - stripping existing exploits, 159–160
 - testing base exploits, 161–162
 - disabling protections, 156–157
 - SEH overwrite exploits, 166–171
- port scanning
 - with Metasploit, 25–26
 - with Nmap
 - compromising Windows machines, 68
 - importing results into Metasploit, 20–22
 - running Nmap from MSFconsole, 24–25
 - running quick scans, 19–20
 - service-enumeration scans, 20
 - TCP idle scans, 22–24
 - SYN, 26, 204–205
- post exploitation, 3–4
 - Meterpreter commands and syntax, 242–243
 - simulated penetration test, 208–211
- PostgreSQL database system, 21
- post/multi/manage/shell_to_meterpreter* module, 103
- powershell_upload_exec function, 180–181
- preengagement interactions, 2, 204
- pre-shared keys, 133, 135
- principals, 220
- privileged labels, module info sheet, 54
- privilege escalation, 77–80
 - attacks, 229–231
- Privilege Escalation Awesome Scripts Suite (PEASS), 230
- processes
 - injection, 72
 - isolation, 225
 - migration, 71
- Process Explorer, 97–98
- proof-of-concept exploits, 158
- protected management frames (PMFs), 134
- protocol fuzzers, 145
- Provided By section, module info sheet, 55
- proxy chains, 209–211
- ProxyChains tool, 210–211
- pry-byebug, 154
- ps command, 80–81, 241
- psnuffle* module, 138–139

PTES. *See* Penetration Testing Execution Standard
pwd command, 65
Python files, generating executables from, 104–105

Q

Quick TFTP Pro 2.1 exploit, 166–171

R

Railgun, manipulating Windows APIs with, 88–89
randomization, adding to exploits, 163, 168–169
rank labels, module info sheet, 54
Rapid7, 6, 26, 37, 49
read-only (RO) community strings, 31
read/write (RW) community strings, 31
reboot command, 241
RECONFIGURE command, 175
references section, module info sheet, 55
reg *command* command, 241
registers, 156
related modules section, module info sheet, 55
Remote Desktop Services, enabling, 84
remote GUIs, accessing, 84, 243
reports, 4
 Nessus
 importing into Metasploit, 45–46
 viewing, 44–45
 Nexpose, importing into Metasploit, 40
resource scripts, 10
return-oriented programming (ROP), 201
rev2self command, 78, 241
reverse_https shell, 104
reverse proxies, 18
reverse shells, 8, 57, 92–93, 103
Rhino Security Labs, 225–226
RHOST option, 23, 240
roles, 220
Rubber Ducky, 118
run command, 59, 150–151
 run post/windows/manage/enable_rdp, 84

S

S3 (Amazon Simple Storage Service) buckets, 29, 222–223
S3Scanner, 29
save command, 59
saving exploit settings, 59

scanner/http modules, 148
scanning Linux systems, 209–211
scan policies, creating in Nessus, 44
scraping systems, 85
screenshot command, 71, 241
screenshots, capturing, 71
search command, 52–53, 99, 148, 240
searching for exploits, 52–55
searchsploit tool, 53–54
Secure Shell (SSH) server version, scanning for, 29–30
SEH. *See* Structured Exception Handler
SEHOP protection, disabling, 157, 191
selecting exploits, 56–59
send_request_cgi method, 151
serverless functions, 221–222
Server Message Block (SMB)
 scanning for, 26–28
 validating logins, 48–49
service-enumeration scans, 20
services command, 25
sessions -c *cmd* command, 240
sessions -K command, 240
sessions -l command, 240
 sessions -l -v, 240
sessions -u command, 77, 103, 207–208, 240
SET. *See* Social-Engineer Toolkit
set command, 58–59, 61, 240
setdesktop *number* command, 241
setg command, 59, 240
set payload command, 240
set target command, 240
shellcode, 8, 11–12
shell command, 62, 77, 241, 243
shikata_ga_nai encoder, 94–96
show advanced command, 146, 240
show auxiliary module, 52, 240
show exploits command, 52, 240
show options command, 52, 180, 240
show payloads command, 57–58, 240
show targets command, 58, 240
-sI flag, 23–24
Simple Network Management Protocol (SNMP), 31–32
simulated penetration tests, 203
 attacking Apache Tomcat, 212–214
 attacking obscure services, 214–215
 covering tracks, 215–217

- simulated penetration tests (*continued*)
 - exploitation, 206–208
 - intelligence gathering, 204–205
 - post exploitation, 208–211
 - preengagement interactions, 204
 - threat modeling, 205–206
- size of payloads, viewing, 194
- smart_hashdump command, 73–74
- SMB. *See* Server Message Block
- SMB Login Check Scanner, 48–49
- smb_login* module, 48
- smb_version* module, 26–28
- sniffer_dump command, 241
- sniffer_interfaces command, 84, 241
- sniffer_start command, 241
- sniffer_stats command, 241
- sniffer_stop command, 241
- sniffing traffic with Metasploit, 138–139
- snmp_enum* module, 31
- snmp_login* module, 31
- Social-Engineer Toolkit (SET), 107
 - Infectious Media Generator attacks, 117–119
 - spear-phishing attacks, 109–113
 - updating and configuring, 108–109
 - web attacks, 113–117
- socket, Docker, 223
- SOCKS protocol, 209–210
- spear-phishing attacks, 109–113, 122
- specialty vulnerability scanners, 48–50
- Sprocket Security, 113
- SQL Server Express, installing, 234–235
- squatting, 109–110
- sS flag, Nmap port scanning, 19–20
- SSH keys, 86–88
- ssh_login_pubkey* module, 87
- SSH (Secure Shell) server version, scanning for, 29–30
- ssh_version* module, 29–30
- stack cookies, 165–166
- stand-alone binaries, creating with MSFvenom, 92–93
- stealthcopter, 231
- steal_token *PID* command, 241
- storage, cloud, 222–223
- stored procedures, 174–175
- Structured Exception Handler (SEH)
 - controlling, 190–192

- overwrites
 - developing, 192–198
 - porting, 166–171
- SurgeMail application, 186–192
- SYN port scanner, 26, 204–205
- sys_exec function, 69
- sysinfo command, 71, 242

T

- tabnabbing, 116
- tail command, 153, 187
- targeted scanning, 26
 - for FTP servers, 30
 - for poorly configured MS SQL servers, 28–29
 - for S3 buckets, 29
 - for Server Message Block, 26–28
 - for Simple Network Management Protocol, 31–32
 - for SSH server version, 29–30
- targets, showing, 58
- TCP scanner, 32–33
 - idle scans, 22–24
- templates, custom executable, 97–98
- terminology, 8
- Terraform command line interface, 225
- test environments
 - operating in, 16
 - setting up, 233
 - ARM and Apple Silicon, 235–236
 - installing Kali metapackages, 236–237
 - x86 and AMD64, 234–235
- testing fuzzers, 187–190
- theHarvester tool, 110
- THREADS value, 23, 27
- threat modeling, 2–3
 - simulated penetration tests, 205–206
- timestamp command, 216, 242
- time to live (TTL), 205
- token impersonation, 80–82
- traffic
 - sniffing with Metasploit, 138–139
 - viewing on targets, 84–85
- Trivial File Transfer Protocol (TFTP), 168
- Twitter, 122
- two-factor authentication, bypassing, 116–117

U

- Ubuntu Machine, exploiting, 63–65
- uictl enable command, 242
- unset command, 58–59
- unsetg command, 59
- upload command, 242
- uploading user-defined functions, 69–70
- UPX packer, 96–97
- USB human interface devices (HIDs), 118–119
- use command, 56, 61, 147, 240
- use incognito command, 81, 242
- use post/linux/manage/sshkey_persistence command, 86
- use priv command, 73, 78, 242
- User Access Control (UAC), 243
- user accounts, privilege escalation for, 77–80
- usernames, harvesting, 113–116
- users, 220
- use sniffer command, 242
- utilities, Metasploit, 11–12

V

- validating SMB logins, 48–49
- VirusTotal, 93, 95, 97–98, 102–103
- Vulnerability and Exploit Database, 49–50
- vulnerability scanning, 3, 35–36
 - basic scans, 36–37
 - with Nessus, 42
 - configuring, 42–43
 - creating scans, 43–44
 - importing results into Metasploit, 45–46
 - scanning in Metasploit, 46–48
 - viewing reports, 44–45
 - with Nexpose, 37
 - configuring, 37–40
 - importing reports into Metasploit, 40
 - running in MSFconsole, 40–42
 - specialty scanners, 48–50
- vulnerable services, identifying, 211–212
- vuIms command, 27–28

W

- web API vulnerabilities, 220
- web attacks, 113
 - bypassing two-factor authentication, 116–117

- tabnabbing, 116
- username and password harvesting, 113–116
- webdav_scanner* module, 147
- website cloning, 113–117
- whoami command, 62, 82
 - whoami /priv, 175–176
- Whois lookups, 16–17
- Wi-Fi attacks, 131
 - capturing and cracking handshakes, 135–136
 - connecting to wireless adapters, 132
 - death and DoS attacks, 133–135
 - Evil Twin attacks, 136–138
 - harvesting credentials with Wi-Fi Pineapple, 139–143
 - monitoring traffic, 132–133
 - sniffing traffic with Metasploit, 138–139
- Wi-Fi Pineapple, 139–143
- wifite* wordlist, 136
- Wi-Fi traffic, monitoring, 132–133
- Windows
 - Active Directory servers, 234
 - APIs, manipulating with Railgun, 88–89
 - Portable Executable, generating, 92
 - User Access Control, bypassing, 243
 - virtual machines
 - compromising, 67–70
 - exploiting, 60–63
- windows_defender_exe* evasion module, 100–101
- windows/smb/psexec* module, 74
- wireless adapters, connecting to, 132
- wireless attacks. *See* Wi-Fi attacks
- Word documents, exploiting, 126–127
- workspace command, 41
- WPA four-way handshakes, 135–136
- Wright, Jordan, 112
- writing Metasploit modules, 173
 - getting command execution on MS SQL, 173–178
 - targeting MS SQL, 178
 - defining exploits, 180
 - editing existing modules, 178–179
 - running exploits, 183–184
 - running shell exploits, 180
 - uploading PowerShell scripts, 181–183

X

X (social media company), 122
x86 architecture, 234–235
x86/shikata_ga_nai encoder, 12
xp_cmdshell procedure, 174–178

Z

Zate, 46
Zphisher, 108
 username and password harvesting,
 113–116